29

INTELLIGENT PROCESS PLANNING AND CONTROL FRAMEWORK FOR THE INTERNET

John P.T. Mo, Stuart Woodman
CSIRO Manufacturing and Infrastructure Technology, Australia
john.mo@csiro.au, stuart.woodman@csiro.au

Abstract: The use of computers has been growing rapidly in manufacturing since computer technology started to be used in machine controls. However, due to historical reasons, these technologies are isolated and their information cannot be shared. For example, to use the information from the CAD system for machining, a human expert engineer has to interpret the CAD meanings and then enters them into a CAM format. Integration of these manufacturing functions requires a technology break-through in automation of manufacturing technology. This paper describes the development of an integrated web-based CIM environment called J-MOPS which can intelligently transform CAD information into machine programs. The use of Java significantly enhanced the system with the advantages of robustness, object-oriented, network oriented, and platform independence.

Key words: Computer automated process planning (CAPP), Integrated computer aided manufacturing (ICAM), Object oriented programming

1. INTRODUCTION

The use of computers has been growing rapidly in manufacturing since computer technology started to be used in machine controls. In the beginning, each manufacturing system's component was developed separately resulting in a term known as *The Islands of Automation* [1,2]. For example, at the design stage in manufacturing, Computer-Aided Design (CAD) has been made feasible after the Cathode Ray Tube (CRT) was invented for generating graphic displays. At the same time, on the shop floor, Computer-Aided Manufacturing (CAM) has been developing, with technologies such as microprocessors for Numerical Control (NC) of machines. These technologies are isolated and their information cannot be shared. To use the information from the CAD system, a human expert engineer has to interpret the CAD meanings and then enters them into a CAM format. Likewise, to improve CAD by experiences in CAM, a human designer has to learn the essence of CAM programming and adjusts the design manually [3]. There is no direct computer path for these two activities to be linked automatically. Each *Island of Automation* was advanced separately until the need to use information from one source on another computer assisted application arose, i.e. Computer Integrated Manufacturing (CIM).

Integration of all manufacturing components requires a technology break-through in automation of manufacturing technology. One of CIM examples is CAPP systems [4]. CAPP helps manufacturing systems to integrate processes from the design to actual manufacturing. Typically, a Part-Program of a CNC machine is generated by an automated program in a CAPP system using data in the CAD model [5]. Although, there are so many CAPP systems nowadays, the development and improvement of CIM is still continuing.

CIM development inevitably involves substantial software development work. In the last decade, a new approach was introduced into computer software development, the object-oriented approach [6,7]. This approach speeds up the development of new programs, and improves the maintainability, reusability, and modifiability of software. Extending this philosophy to manufacturing, CIM system also adopts the object-oriented concept. Experience has shown that this concept brings significant improvement in manufacturing technology [8,9]. For example, Manufacturing Objects Processing System (MOPS), which is a framework for CAD/CAM system using object-oriented concepts, was developed in C++ programming language [10]. The framework is an intelligent manufacturing system for a complete integration from design (CAD) to actual manufacturing of a product (CAM) [11].

2. SYSTEM REQUIREMENTS

Recent development in computer networks has revealed that a computer is not just a single unit tool, but is also a part of a complex network with other computers. An incompatible system amongst computers is a common problem for an integration system. Every different computer system has its own platform of software and hardware. This prevents information being transferred across the system boundary. For example, a PC compatible application cannot run in a UNIX system and vice versa. This will be a great obstacle for now and in the future, when the Internet and network play an important role in computer communication linking different computer systems or platforms electronically [12].

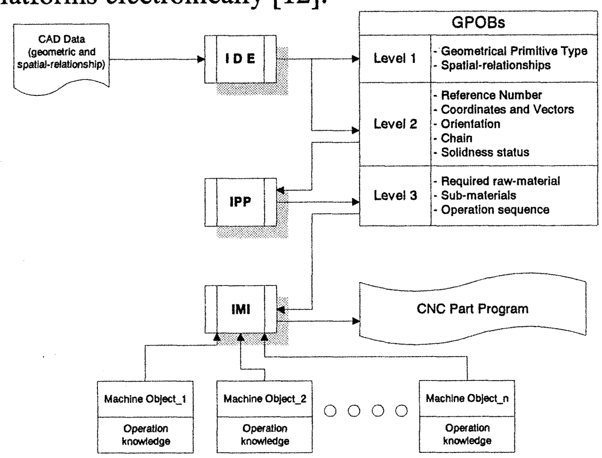

Figure 1: Structure of MOPS.

The widespread use of computer networks and especially the availability of the Internet has a significant impact on application accessibility [13,14]. Applications which are tied to a particular platform are therefore losing out on the general usability. Many applications were designed for single-user single-platform and this prohibits people on other computer environments from access. For example, Manufacturing Objects Processing System (Figure 1) depends on PC environment and cannot be moved onto the World Wide Web.

3. JAVA MANUFACURING OBJECT PROCESSING SYSTEM

This paper describes the development of an environment with significant enhancement in graphical and web-based delivery capabilities. The system, which is called J-MOPS (Java MOPS), is created using Java language. Java has advantages of robustness, object-oriented, network oriented, and platform independence [15]. J-MOPS integrates the design stage to the manufacturing activities in the NC machine, showing the examined functions and elements of the objects. J-MOPS is developed with an aim to be moved to the World Wide Web and hence it is independent of hardware or operating system, thereby providing the biggest possible usage in the modern global CIM era. J-MOPS incorporates the object-oriented philosophy of MOPS and extends its graphical functionality. Geometrical Primitive Objects (GPOBs) received from a CAD data system are extracted according to their geometrical characteristics and relationships. Then the extracted data is processed in an object-oriented approach into a detailed description of the next manufacturing processes and machine programs.

J-MOPS uses the latest Java3D [16] technology to implement its graphics modelling of the manufactured and raw material product in 3D display. The result is a completely new 3D network-enabled graphical computer aided process planning (CAPP) system, which can be accessed by any web browser. Using the interactive 3D viewing capability, users can rotate, zoom, move the objects on the screen as if they are held manually. The effect is a much better visualisation of the product being designed and hence eliminates as many design errors as possible.

It is obvious that the prototype and model of the actual real world plays an important role for helping decision making in industries. The approach of bringing the real objects in a three-dimensional modelling graphical display is one of the aims of the research. The other improvement is to move the framework into a more user-friendly application. Using the Java language, J-MOPS provides a platform independent system similar to MOPS which can be accessible from any place in the world using a web browser. The system is an expansion of an earlier version in FORTRAN and C [17]. In the last couple of years, latest software design technologies have been applied to improve the modularity and connectivity of the system [18,19].

The system uses the concept of three levels data abstraction, called Geometrical Primitive Object (GPOB). The GPOB is a universal set of objects that can interact with processing modules. The first level of GPOBs consists of objects with basic geometrical attributes. This information is retrieved from the CAD data model when it was loaded by the system. In the further processing, the CAD data models are extracted and assembled into the second level of GPOBs, consisting of attributes such as the information of coordinates, vectors, orientation, the relationships to other object, and the nature of solidness.

In actual implementation, the system consists of three main modules: Intelligent Data Extractor (IDE), Intelligent Process Planning (IPP), and Intelligent Machine Interface (IMI); and one additional module for three-dimensional graphical display. From the CAD system, which is the input to the system, until the CNC Part-Program result, which is the output of the system, the data are processed object by object. In the system, all of these modules are acting as objects themselves.

4. THE INTELLIGENT DATA EXTRACTOR

Intelligent Data Extractor (IDE) is the entry module to activate the system. Its aims are to extract CAD data models and to assemble them as GPOB information for the input of next module. The CAD data model, retrieved by IDE, supplies information for GPOBs level one: geometrical primitive type and spatial-relationships (Figure 2). Then, IDE extracts this information to produce GPOBs level two, such as the reference number, vertices and vectors, volume, orientation, spatial-relationships, and object's nature as solid or hollow object.

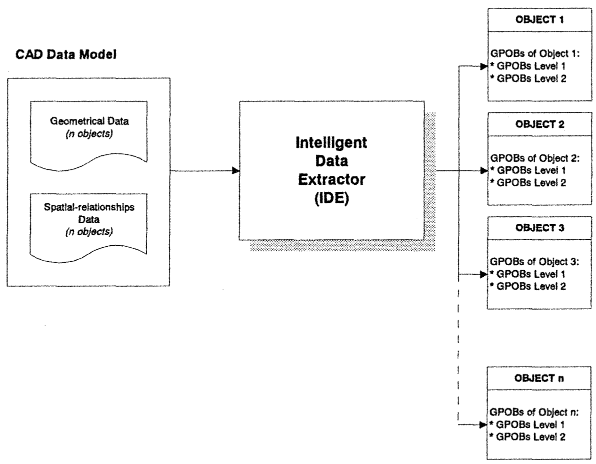

Figure 2: GPOBs in Intelligent Data Extractor.

The nature of the object, in term of its solidness, is determined by analysing chains among GPOBs. Using an algorithm developed by Liang et al [4], IDE determines the chains, which will be used to identify the solidness of the objects. The algorithm consists of iterations of chain list and utilises a series of postulates and reasoning to decide the characteristics of the objects in the product. All of the GPOBs, assembled by IDE, are used for further manipulation in the next modules. Thus, the information assembled by IDE has to present informative data for smooth transforming of the CAD model data to the actual manufacturing data.

5. THE INTELLIGENT PROCESS PLANNER

The second module of the system is called Intelligent Process Planner (IPP). This module receives and processes the GPOBs information from IDE in order to produce an operation sequence needed for the manufacturing activities. This sequence will be used for the guideline operation of CNC machine on the shop floor. IPP works in three stages of operation. The activities are to transform the information from IDE into material stock requirements and operation sequence in manufacturing.

IPP consists of several sub-modules in order to accomplish those three levels of operation. The first sub-module is the Information Interpreter. This sub-module is responsible for receiving any information coming as input for IPP and interpreting this information, so it is ready to be used and processed in other sub-modules. The second module, the Data Share Stock, is the central database for IPP. It is responsible for receiving and providing all the information from and to the sub-components in IPP. The third sub-module is the Material Generator. This sub-module works to accomplish the second stage of IPP operation. It consists of two components: Master Material Generator and Sub-Material Generator. These components have the role of generating base material and sub-materials, which are the required materials for manufacturing the part objects. There are algorithms and some rules in order to generate the base material and sub-materials.

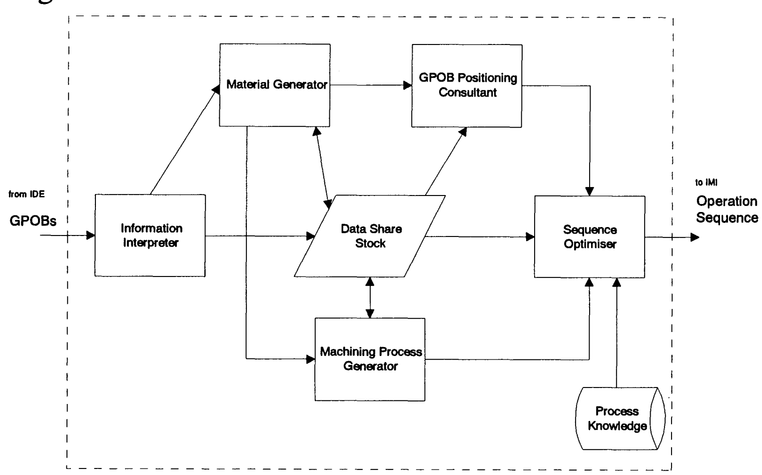

Figure 3: Sub-modules of IPP

Using the same philosophy of IDE, IPP also generates the base material and each of sub-materials in its own GPOB properties. The nature of the base material and sub-material is made as a hollow object. The reason is because the materials contain the manufactured part objects, which are solid objects.

The next two sub-modules, the Machining Process Generator and the Sequence Optimiser, are the important sub-modules for the last stage operation in IPP. The operation sequence for machining is generated by the

Machining Process Generator. In general, there are three considerations for generating the operation sequence.

The processes in the operation sequence are the removal action. It is possible because the GPOBs and sub-materials are objects. Then the sequence is passed to the Sequence Optimiser to be optimised. Based on the knowledge base and information from the GPOB Positioning Consultant sub-module, the sequence from the Machining Operation Identifier is updated.

6. INTELLIGENT MACHINE INTERFACE

Intelligent Machine Interface (IMI) is the third module in the system. This module generates part programs for machining processes on the shop floor. IMI consists of machine objects, which contain the knowledge base of controlling the manufacturing machine.

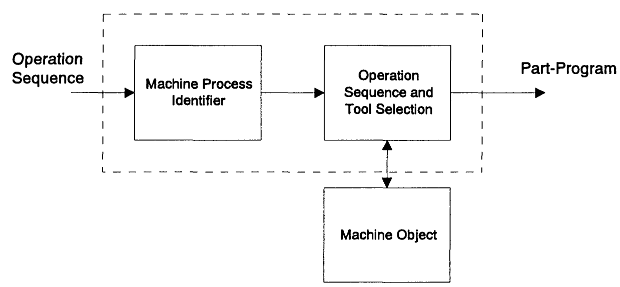

Figure 4: Sub-modules of IMI.

IMI has two sub-modules: Machine Process Identifier and Operation Sequence and Tool Selection. It also uses the knowledge base of controlling machines, which is located in the Machine Objects. The Machine Process Identifier identifies and groups each machining process information of the operation sequence, received from IPP, into some machining process categories, such as milling, drilling, and grinding, so that same category of machining processes can be sent to the right machine for manufacturing operation. However, in this system the machining process is still limited only to the CNC milling machine.

After that, the Operation Sequence and the Tool Selection will work to assign the machining process sequence and the tool selection for each of the machining process categories in every machine, using the knowledge base of machines provided by the machine object. Then, a part program is ready to be generated with the GPOBs as its referred information.

7. THE JAVA 3D GRAPHIC MODULE

Three-dimensional Graphic Display is the last module in the system. It presents the final manufactured parts and the raw materials as three-dimensional graphical objects and interacts with the user to give the best viewing. The module consists of three sub-modules: Object Recogniser, Graphical Object Generator, and Appearance Properties. The module draws the final manufactured parts using the GPOBs from the output of IDE, whereas the raw materials will be drawn according to their GPOBs received from IPP. The Object Recogniser receives the GPOBs and distinguishes the object to produce necessary information, such as: material or part, hollow or solid object, cube or cylinder, etc.

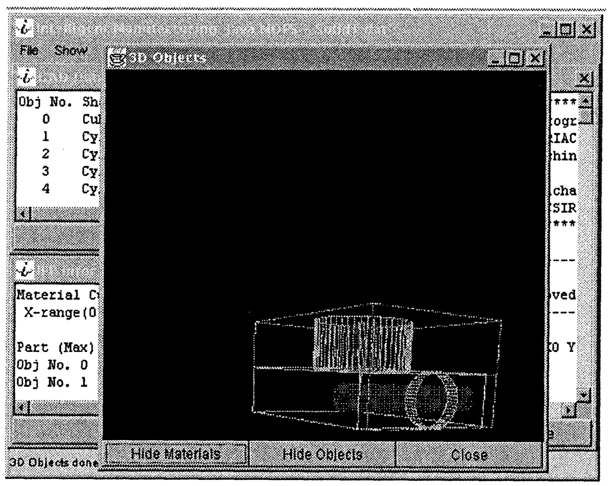

Figure 5: 3D Display in wire-frame mode.

The module works in object-oriented philosophy, which means that every object is drawn individually as an object. The object is created using a suitable class, chosen according to its characteristic from the Graphical Primitive Objects. The third sub-module is Appearance Properties. This sub-module supplies the appearance properties such as colour and drawing mode to the Graphical Object Generator, so it can generate the object in particular appearance according to the information received from Object Recogniser.

Using the inherent capability of Java 3D, the system can be easily programmed to display the objects in wire-frame mode (Figure 5) or rendered mode (Figure 6).

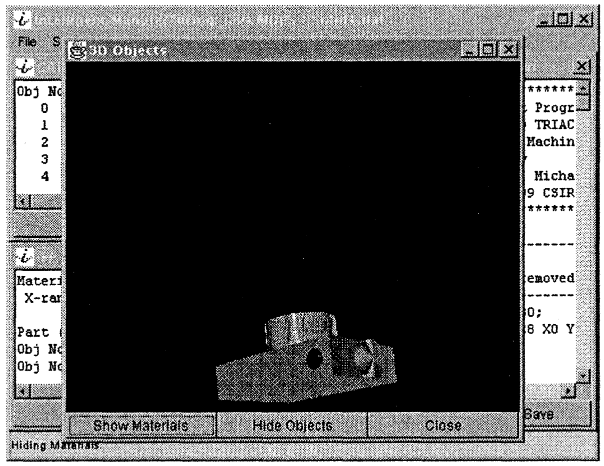

Figure 6: Display in rendered mode.

8. SUMMARY

J-MOPS is developed based on MOPS philosophy. It represents a major advancement of offering the complete CAPP and 3D simulation capabilities on the World Wide Web. It simplifies the system requirements for the user and removes the dependency on platforms. The results of this study provide the basis for supporting design to be done intelligently on a global basis.

REFERENCES

1. Chakravarty A.K. Dimensions of Manufacturing Automation, Int. J. Prod. Res., Vol.25, No. 9, 1987, pp.1339-1354
2. Harhalakis G., Lin C.P., Hillion H., Moy K.Y., Development of a Factory Level CIM Model, *The Journal of Manufacturing Systems*, Vol.9, No. 2, 1990, pp.116-128
3. Lin A.C., Wang H.P.B. The Development of a Computer-Integrated Design and Manufacture System for Plate Cams, *Manufacturing Review*, Vol.1, No.3, 1988, October, pp.198-213
4. Subrahmanyam S., Wozny M., An Overview of Automatic Feature Recognition Techniques for Computer-aided Process Planning, *Computers in Industry*, Vol.26, 1995, pp.1-21
5. Liu C.H., Perng D.B., Chen Z., Automatic Form Feature Recognition and 3D Part Reconstruction From 2D CAD Data, *Computers & Industrial Engineering*, Vol.26, No. 4, 1994, pp.689-707
6. Ainger A.W.S., Schmid F., The Helical Approach to Software Design, *Journal of Computer Integrated Manufacturing Systems*, Vol.8, No. 2, 1995, May, pp.105-115

7. Troll M., Brandon J.A.,Vollmer E., Object-oriented Methodology for Optimizing Information Flow in Manufacturing, *International Journal of Flexible Automation and Integrated Manufacturing*, Vol.4, No. 3&4, 1996, pp.137-153
8. Westfechtel B., Integrated Product and Process Management for Engineering Design Applications, *Integrated Computer-Aided Engineering*, Vol.3, No. 1,1996, pp.20-35
9. Usher J.M., A STEP-Based Object-Oriented Product Model for Process Planning, *Computers and Industrial Engineering*, Vol.31, No. 1/2, 1996, pp.185-188
10. Mo J.P.T., Fernandez G., Liang G., "An Object Oriented Framework for Intelligent Manufacturing from Computer Aided Design Features", ETCE '93, Computers in Engineering Symposium, ASME, 31 January, 1993, Houston, USA
11. Mo, J.P.T. and Ranganathan, D., From Design to Manufacture by Objects, *International Journal of Flexible Automation and Integrated Manufacturing*, 4(2): 95-110, 1996.
12. Nemes L., Bertok P., Cooperative Design on the Internet,IFIP World Congress on IT Tools, 2-6 September, 1996, IFIP - International Federation for Information Processing, Canberra, Australia, pp.419-428
13. Bertok P., Mo J.P.T., "Collaboration and Application Integration: Distributed Design with Virtual CAD", 4th International Conference on Design of Information Infrastructure Systems for Manufacturing, 15-17 November, 2000, Melbourne
14. Jiang H.C., Mo. J.P.T., "Computer Aided Design over the Internet", 26th International Conference on Computers and Industrial Engineering, 15-17 December, 1999, Melbourne, Australia, Vol.1, pp.497-501
15. Au, E. and Makower, D., Java Programming Basics, MIS Press, New York, 1996.
16. Sowizral, H., Rushforth, K. and Deering M., The Java 3D API Specification, Sun Microsystems, California, 1998.
17. Liang, G., Fernandez, G. and Mo, J.P.T., Intelligent CAD Data Recognition for an Object-Oriented CAD/CAM System, Proceedings of International Conference on Data and Knowledge Systems for Manufacturing and Engineering, Hong Kong, 2-4, 7-12 May 1994.
18. Ooi S.T., Bertok P., Zhao L.P., Jiang H.C., Mo J.P.T., "Patterns as Reengineering Front-End in Distributed Environments", *Proceedings of 2nd Australian Workshop on Software Architectures*, 23 November, 1999, Melbourne, Australia, pp.75-82
19. Bertok P., Mo J.P.T., "Software Technology for Design System Integration", 4th International Conference on Design of Information Infrastructure Systems for Manufacturing, 15-17 November, 2000, Melbourne